

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

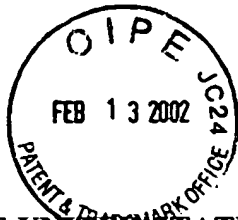
Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problem Mailbox.**



COPY OF PAPERS  
ORIGINALLY FILED

2673 #2

**PATENT**

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

Applicant: Heui-Do Lim

Docket: 678-642 (P9697)

Serial No: 10/038,042

Date: January 29, 2002

Filed: January 3, 2002

For: **METHOD AND APPARATUS  
FOR RECOGNIZING KEY**

**RECEIVED**

**FEB 20 2002**

Assistant Commissioner for Patents  
Washington, D.C. 20231

Technology Center 2600

**TRANSMITTAL OF PRIORITY DOCUMENT**

Sir:

Enclosed is a certified copy of Korean Appln. No. 1263/2001 filed  
on January 10, 2001 from which priority is claimed under 35 U.S.C. §119.

Respectfully submitted,

Paul J. Farrell  
Registration No. 33,494  
Attorney for Applicant

**DILWORTH & BARRESE, LLP**  
333 Earle Ovington Boulevard  
Uniondale, New York 11553  
(516) 228-8484

PJF:cm

**CERTIFICATE OF MAILING UNDER 37 C.F.R. § 1.8 (a)**

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail, postpaid in an envelope, addressed to the: Commissioner of Patents and Trademarks, Washington, D.C. 20231 on January 29, 2002

Dated: January 29, 2002

  
Paul J. Farrell



# 대한민국 특허청

KOREAN INTELLECTUAL  
PROPERTY OFFICE

RECEIVED  
FEB 20 2002  
Technology Center 2600

별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto  
is a true copy from the records of the Korean Intellectual  
Property Office.

출원 번호 : 특허출원 2001년 제 1263 호  
Application Number

출원 년 월 일 : 2001년 01월 10일  
Date of Application

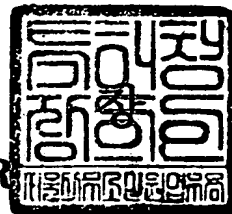
출원인 : 삼성전자 주식회사  
Applicant(s)



2001 년 03 월 21 일

특 허 청

COMMISSIONER



【서류명】	특허출원서
【권리구분】	특허
【수신처】	특허청장
【참조번호】	0001
【제출일자】	2001.01.10
【국제특허분류】	H04Q
【발명의 명칭】	키 인식방법 및 장치
【발명의 영문명칭】	METHOD AND APPARATUS FOR RECOGNIZING KEY
【출원인】	
【명칭】	삼성전자 주식회사
【출원인코드】	1-1998-104271-3
【대리인】	
【성명】	이건주
【대리인코드】	9-1998-000339-8
【포괄위임등록번호】	1999-006038-0
【발명자】	
【성명의 국문표기】	임희도
【성명의 영문표기】	LIM,Heui Do
【주민등록번호】	700210-1774512
【우편번호】	442-371
【주소】	경기도 수원시 팔달구 매탄1동 173-92 천일하이츠 201호
【국적】	KR
【심사청구】	청구
【취지】	특허법 제42조의 규정에 의한 출원, 특허법 제60조의 규정에 의한 출원심사를 청구합니다. 대리인 이건주 (인)
【수수료】	
【기본출원료】	19 면 29,000 원
【가산출원료】	0 면 0 원
【우선권주장료】	0 건 0 원
【심사청구료】	9 항 397,000 원
【합계】	426,000 원
【첨부서류】	1. 요약서·명세서(도면)_1통

**【요약서】****【요약】**

본 발명은, 키 매트릭스 구조를 가지는 단말장치에서의 키 인식방법에 있어서, 상기 키 매트릭스 상의 이전 키 상태 값과 키 상태 변화에 따른 현재 키 상태 값과의 차이를 이용하여 키 누름과 키 떨어짐을 파악하는 과정과, 상기 키 매트릭스를 구성하는 행들과 열들 각각에 대응 비트가 할당되어 있고, 상기 행들 각각에서 인접한 열들간 차이 값과 상기 열들 각각에서의 인접한 행들간 차이 값을 고려한 관계식과 상기 이전 키 상태 값과 현재 키 상태 값간의 차이를 이용하여 키 매트릭스 상에서의 입력 키 위치 값을 구하는 과정으로 이루어진다.

**【대표도】**

도 4

**【색인어】**

키 인식, 키 매트릭스

**【명세서】****【발명의 명칭】**

키 인식방법 및 장치{METHOD AND APPARATUS FOR RECOGNIZING KEY}

**【도면의 간단한 설명】**

도 1은 본 발명의 실시 예에 적용되는 블록 구성도,

도 2는 비트별 키 매트릭스를 보여 주는 도면,

도 3은 키 매트릭스의 키 인터럽트 결과를 표시하는 내부 레지스터 구성도,

도 4는 본 발명의 실시 예에 따른 키 인식 제어 흐름도.

**【발명의 상세한 설명】****【발명의 목적】****【발명이 속하는 기술분야 및 그 분야의 종래기술】**

- <5> 본 발명은 입력장치에 관한 것으로, 특히 키 매트릭스 구조의 키패드나 키보드의 키를 인식하는 방법에 관한 것이다.
- <6> 요즈음 널리 사용되고 있는 전자 제품들 예컨대, 컴퓨터나 휴대용 이동단말들은 대부분이 키 매트릭스 구조의 키패드나 키보드를 키 입력장치로서 구비한다. 키 매트릭스 구조의 키 입력장치들을 구비한 단말 제어부는 키 인터럽트 방식이나 폴링방식으로 사용자의 키 입력을 인식한다. 키 인터럽트 방식에서는 키패드나 키보드상의 키 상태 변화가 있게 되면 그것이 키 인터럽트로서 단말 제어부로 발생되고 단말 제어부는 그에 따라

키 인식을 하게된다. 폴링방식은 단말 제어부가 일정한 주기로 키 상태 변화가 있는지를 체크한다. 키 인터럽트 방식에서는 키 인터럽트가 발생하면 제어부는 어떤 키의 변화가 키 인터럽트를 발생시켰는지 알기 위하여 내부 레지스터의 각 비트별로 체크하여 현재의 키 상태를 확인한다. 그리고 이전의 키 상태와 비교하여 어떤 키가 변화하였는지도 확인한다.

<7> 이를 도 2에 도시된 비트별 키 매트릭스 구조 및 도 3에 도시된 키 매트릭스의 키 인터럽트 결과를 표시하는 내부 레지스터 구성을 참조하여 보다 상세히 설명하면 하기와 같다. 도 2 및 도 3의 키 매트릭스 구조의 일 예는 5x5구조의 키 매트릭스 구조이다. 키 매트릭스상에 있는 어느 키도 눌러지 않은 상태에서는(초기 상태) 내부 10비트 레지스터의 비트 b0에서 b9까지가 모두 '0'이다. 즉 키 매트릭스로부터의 출력 결과 값은 '00000 00000'이다. 이 때 사용자에게 의해 키 매트릭스상의 0행 0열의 키 즉 key[0][0]이 눌러지게 되면 키 인터럽트가 발생되고 그에 따라 내부 10비트 레지스터에 표시되는 인터럽트 결과 값은 '00001 00001', 즉 0x21(H: hexa decimal)이다. 여기에서 0행과 0열의 키 key[0][0]이 눌린 것을 확인하기 위하여 제어부는 내부 10비트 레지스터를 상위 5비트와 하위 5비트로 구분하여 각각에서 어떤 비트가 '1'인지를 체크한다. 그래서 현재 키가 눌러 있는 상태가 key[0][0]인 것을 확인한다. 그 후 이전의 키 상태와 비교한다. 이전에는 아무 키도 눌러지지 않았기 때문에 현재 눌린 키가 key[0][0]임을 인식하게 된다. 이러한 상태에서 사용자가 key[1][1]을 누르게 되면 키 매트릭스로부터의 키 인터럽트에 의해 10비트 레지스터에 표시된 키 인터럽트 결과 값은 '00011 00011', 즉 0x63(H)

이다. 여기에서 상위 5비트와 하위 5비트로 나누어서 1인 비트를 확인하게 되면 제어부에 의해서 예상되는 키는  $key[0][0]$ ,  $key[0][1]$ ,  $key[1][0]$ ,  $key[1][1]$ 이다. 이전에  $key[0][0]$ 을 눌렀고 현재 상태에서  $key[1][1]$ 을 눌렀는데 제어부는 이전의 키 상태와 현재의 키 상태를 논리적으로 판단하여  $key[0][0]$ ,  $key[0][1]$ ,  $key[1][0]$ ,  $key[1][1]$  중 현재 눌러진 키가  $key[1][1]$ 임을 찾아내야 한다.

【발명이 이루고자 하는 기술적 과제】

- <8>        상술한 바와 같은 종래 기술은 하기와 같은 단점이 있었다. 첫 번째는 키 매트릭스로부터 매번 키 인터럽트가 발생할 때마다 제어부는 각 비트의 상태를 확인해야 한다. 두 번째는 제어부는 각 비트의 상태를 확인하여 가능한 경우의 수를 설정하여야 한다. 세 번째는 이전의 키 상태를 모두 저장하고 있어야 한다. 네 번째는 이전의 키 상태와 현재 가능한 경우의 모든 키 상태와 비교하여 논리적으로 어떤 키의 변화가 키 인터럽트를 발생시켰는지 알아내야 한다.
- <9>        따라서 본 발명의 목적은 키 입력에 따른 키 위치 값을 쉽게 인식할 수 있는 방법을 제공하는데 있다.
- <10>       본 발명의 다른 목적은 키 매트릭스로부터 키 인터럽트가 발생하면 키 인터럽트 그 자체의 값으로 어떤 키의 변화가 키 인터럽트를 발생시켰는지 산술적으로 계산하여 알아낼 수 있는 방법을 제공하는데 있다.
- <11>       상기한 목적에 따라, 본 발명은, 키 매트릭스 구조를 가지는 단말장치에서의 키 인식방법에 있어서, 상기 키 매트릭스 상의 이전 키 상태 값과 키 상태 변화에



따른 현재 키 상태 값과의 차이를 이용하여 키 누름과 키 떨어짐을 파악하는 과정과, 상기 키 매트릭스를 구성하는 행들과 열들 각각에 대응 비트가 할당되어 있고, 상기 행들 각각에서 인접한 열들간 차이 값과 상기 열들 각각에서의 인접한 행들간 차이 값을 고려한 관계식과 상기 이전 키 상태 값과 현재 키 상태 값간의 차이를 이용하여 키 매트릭스 상에서의 입력 키 위치 값을 구하는 과정으로 이루어짐을 특징으로 한다.

- <12> 또한 본 발명은, 키 매트릭스 구조를 가지는 단말장치에서의 키 인식방법에 있어서, 상기 키 매트릭스 상의 이전 키 상태 값과 키 상태 변화에 따른 현재 키 상태 값이, 상기 현재 키 상태 값 - 이전 키 상태 값 =  $2^{j+5} + 2^i$ 가 되는 관계를 이용해서 변수  $i, j$ 를 계산하는 과정과, 상기 변수  $i, j$ 로 키 매트릭스 상에서 입력 키 위치 값  $key[i][j]$ 임(여기서  $i$ 는 키 매트릭스의 행의 위치이고  $j$ 는 키 매트릭스의 열의 위치)을 구하는 과정으로 이루어짐을 특징으로 한다.

### 【발명의 구성 및 작용】

- <13> 이하 본 발명의 바람직한 실시 예들을 첨부한 도면을 참조하여 상세히 설명한다. 도면들 중 동일한 구성요소들은 가능한 한 어느 곳에서든지 동일한 부호들로 나타내고 있음에 유의해야 한다. 또한 본 발명의 요지를 불필요하게 흐릴 수 있는 공지 기능 및 구성에 대한 상세한 설명은 생략한다.
- <14> 도 1은 본 발명의 실시 예에 적용되는 블록 구성도로서, 키 매트릭스(2), 제어부(4), 램(6) 및 롬(8)을 포함하고 있다. 도 2는 도 1의 키 매트릭스(2)의 비트별 키 매트릭스 구조를 보여주는 도면으로서, 5x5 키 매트릭스 구조를 일 예로 하고 있다. 도 3은

키 매트릭스(2)의 키 인터럽트 결과를 표시하는 제어부(4)의 내부 레지스터 구성도이다. 도 3의 내부 레지스터는 도 2에 일 예로 도시된 5x5 키 매트릭스의 열과 행을 나타내기 위한 상위 5비트와 하위 5비트 구성의 10비트 레지스터이다.

<15>       사용자에 의한 키 입력이 있으면 키 매트릭스(2)의 키 상태가 변화가 제어부(4)에 키 인터럽트 발생으로서 제공되며, 인터럽트 결과 값은 제어부(4)의 도 3에 도시된 바와 같은 내부 레지스터에 표시된다. 제어부(4)는 내부 레지스터에 표시된 키 인터럽트 결과 값에 의거해서 키 매트릭스(2)상의 입력된 키 위치 및 키 상태 변화를 인식하는 동작을 수행한다. 램(6)은 본 발명의 실시 예에 따라 이전 키 상태값을 임시 저장하기 위한 이전(Previous) 레지스터(이하 'P레지스터'라 칭함)와 현재 키 상태 값을 임시 저장하기 위한 현재(Current) 레지스터(이하 'C레지스터'라 칭함)를 포함하고 있다.

<16>       도 4는 본 발명의 실시 예에 따른 키 인식 제어 흐름도이다. 본 발명의 실시 예에서는 도 2와 같은 키 매트릭스 구조에서 키 매트릭스(2)상의 키 상태 변화로 인해 키 인터럽트가 발생할 때, 도 3에 도시된 내부 레지스터에 표시된 인터럽트 결과 값 자체만으로 어떤 키의 변화가 일어났는지를 확인할 수 있다.

<17>       이를 위해서 본 발명의 실시 예에서는 하기 표 1을 참조하여 도 2와 같은 키 매트릭스에서 행렬의 규칙성을 이해하고, 산술적 계산만으로 현재 입력된 키 위치 값을 구하는 관계식을 정의한다.

<18>

【표 1】

SUM = 3E0	b5(0x20)	b6(0x40)	b7(0x80)	b8(0x100)	b9(0x200)
SUM = 1F					
b0(0x01)	0x21(33)	0x41(65)	0x81(129)	0x101(257)	0x201(513)
b1(0x02)	0x22(34)	0x42(66)	0x82(130)	0x102(258)	0x202(514)
b2(0x04)	0x24(36)	0x44(68)	0x84(132)	0x104(260)	0x204(516)
b3(0x08)	0x28(40)	0x48(72)	0x88(136)	0x108(264)	0x208(520)
b4(0x10)	0x30(48)	0x50(80)	0x90(144)	0x110(272)	0x210(528)

- <19> 표 1에서 '0x21(33)'일 예에서 '0x21'값은 16진수(hexa decimal) 표현이고, 괄호()내 값 '33'은 10진수(decimal) 표현이다.
- <20> 상기 표 1에서 본 발명자는 키 매트릭스(2)에서의 행 규칙성과 열 규칙성을 하기와 같이 얻을 수 있다.
- <21> (a) 행 규칙성
- <22> 같은 열에서 인접한 행들간 차이 값(비트 가중치)은 각각 1, 2, 4, 8, 16이다. 이는 각각  $2^0$ ,  $2^1$ ,  $2^2$ ,  $2^3$ ,  $2^4$ 으로 표현될 수 있다.
- <23> (b) 열 규칙성
- <24> 같은 행에서 인접한 열들간 차이 값(비트 가중치)은 각각 32, 64, 128, 256, 512이다. 이는 각각  $2^{5+0}$ ,  $2^{5+1}$ ,  $2^{5+2}$ ,  $2^{5+3}$ ,  $2^{5+4}$ 으로 표현될 수 있다.
- <25> 상기 행 규칙성과 열 규칙성을 이용하면 해당 행과 열이 마주치는 키 위치 값  $key[i][j]$ 는 하기와 같은 관계식으로 정의된다.
- <26> <<관계식>>
- <27>  $key[i][j] = 2^{5+j} + 2^i$
- <28> 여기서 i와 j는 키 상태 변화된 키 위치값을 알려주는 키 매트릭스상의 행과 열을

가리키는 변수이다. 상기 관계식을 이용한 키 위치 값의 일 예를 들면,  $\text{key}[0][0] = 2^{5+0} + 2^0 = 0x21(33)$ 이다.

<29> 이하 도 1 내지 도 4를 참조하여 본 발명의 실시 예에 따른 동작을 상세히 설명한다.

<30> 지금 단말에 전원이 들어오면 제어부(4)는 도 4의 100단계로 진행하여 키 매트릭스(2)상의 키 상태를 확인하여 초기 상태의 키 상태 값인 '00 00000 00000' 즉 0x00(H)(이 값은 초기에 아무런 키도 눌러지지 않은 경우이다)을 램(6)의 P레지스터에 저장한다. 이 후 102단계로 진행하여 키 인터럽트가 발생하는지를 판단한다. 예컨대, 사용자에게 의해  $\text{key}[0][0]$ 이 눌러지면 키 매트릭스(2)로부터 키 인터럽트가 발생한다. 이에 따라 제어부(4)는 104단계로 진행하여 상기 키 인터럽트에 따른 현재 키 상태 값 '00 00001 00001', 즉 0x21(H)을 램(6)의 C레지스터에 저장한다. 그 후 제어부(4)는 C레지스터에 저장된 현재 키 상태 값에서 P레지스터에 저장된 이전 키 상태 값을 빼서 그 값이 양수인지 음수인지를 판단한다. 이러한 판단에 의해서 제어부(4)는 키 인터럽트가 키 눌러짐에 의해 발생했는지 아니면 키 떨어짐에 의해 발생했는지를 알 수 있다. '현재 키 상태 값 - 이전 키 상태 값'의 결과 값이 양수이면 키 인터럽트는 키 눌러짐에 의한 것이고, 그 결과 값이 음수이면 키 인터럽트는 키 떨어짐에 의한 것이다. 제어부(4)는 키 눌림 또는 키 떨어짐의 정보를 이용하여 그에 상응하는 동작을 수행할 수 있다. 예컨대, 1번키가 눌린 상태에서 2번키가 눌린 후 바로 떨어졌다면 본 발명의 실시 예에서는 이러한 키 눌림 및 키 떨어짐을 파악할 수 있고 그에 상응하는 동작을 수행할 수 있다. 상기 일 예에서는 '현재 키 상태 값 - 이전 키 상태 값' =  $0x21(H) - 0x00(H) = 0x21(H)$  즉 양수이므로 키가 눌러져서 발생한 인터럽트임을 알 수 있다.

- <31> 그 후 제어부(4)는 도 4의 108단계로 진행한다. '현재 키 상태 값 - 이전 키 상태 값' =  $0x21(H) - 0x00(H) = 0x21(H) = 2^5 + 2^0 = 2^{j+5} + 2^i$ 의 관계를 이용해서 변수  $i$ 와  $j$ 를 계산한다. 상기 계산된  $i$ 와  $j$ 는 키 상태 변화된 키 위치값을 알려주는 키 매트릭스상의 행과 열을 가리킨다.
- <32> 상기  $i$ 와  $j$ 를 계산함에 따라 제어부(4)는 도 4의 110단계로 진행하여 키 상태 변화된 키 위치 값을  $key[i][j]$ 를 이용하여 구한다. 도 4의 108단계 내지 110단계의 과정은 본 발명이 키 인터럽트 결과 값 자체만으로 어떤 키의 변화가 일어났는지를 확인할 수 있게 해 준다. 상기의 일 예에서는  $0x21(H) = 2^5 + 2^0 = 2^{j+5} + 2^i$  관계에 의해서  $i$ 와  $j$ 는 모두 '0'이고, 키 위치 값  $key[i][j] = key[0][0]$ 이 된다. 그리고 도 4의 106단계에서 '현재 키 상태 값 - 이전 키 상태 값'의 결과 값이 양수이었으므로 키가 눌러짐에 의해 키 인터럽트가 발생한 것임을 알 수 있다. 정리하면 상기 일 예에서는 키 인터럽트가 키  $key[0][0]$ 이 눌러짐에 발생할 것임을 알 수 있다.
- <33> 그 후 제어부(4)는 다음의 키 상태 변화를 감지하기 위해 C레지스터에 저장된 현재 키 상태 값을 P레지스터에 이전 키 상태 값으로서 저장한다. 그 후 102단계로부터의 과정을 다시 수행한다.
- <34> 예컨대, 다음에 눌러지는 키가  $key[1][1]$ 이면, 현재 키 상태 값은 =  $0x63(H)$ 이고 이전 키 상태 값은 =  $0x21(H)$ 이다. 현재 키 상태 값 - 이전 키 상태 값 =  $0x63(H) - 0x21(H) = 0x42(H) = 2^6 + 2^1 = 2^{j+5} + 2^i$ 이므로, 가중치  $i, j$ 는 모두 1이다. 그에 따라 키 매트릭스상에서  $key[1][1]$ 이 눌러졌음을 알 수 있다. 한편  $key[0][0]$  다음에  $key[0][1]$ 이 눌러졌다고 가정하면, 현재 키 상태 값은 =  $0x61(H)$ 이고 이전 키 상태 값은 =  $0x21(H)$ 이다. 현재 키 상태 값 - 이전 키 상태 값 =  $0x63(H) - 0x21(H) = 0x40(H) =$

$2^6 = 2^{j+5} + 2^i$ 으로서  $j$ 는 1인데  $i$ 값은 구할 수가 없다. 이 경우는 같은 행에서 키가 눌러진 경우이다. P레지스터에 저장된 이전 키 상태 값은  $0x21 = 2^5 + 2^0$ 로서 행의 위치가 0이므로,  $key[0][1]$ 이 눌러졌음을 알 수 있다.

<35> 상기한 바와 같이 본 발명에서는 키 상태가 변한 키의 위치를 단순히 가중치를 두어 표현함으로써 숫자 자체로 키의 위치를 표시할 수 있다. 또한 키의 상태가 변했을 때 현재의 키 상태 값을 이전 키 상태 값과 비교함으로써 키가 눌렸는지 떨어졌는지를 판단할 수 있다.

<36> 상술한 본 발명의 설명에서는 키 인터럽트 방식 및 5x5 키 매트릭스 구조의 일 예를 들어 구체적인 실시 예로서 설명하였지만 폴링방식 및  $k \times m$  ( $k, m$ 은 자연수) 키 매트릭스 구조로의 변형이 본 발명의 정신을 벗어나지 않는 범위 내에서 실시할 수 있다. 따라서 본 발명의 범위는 설명된 실시 예에 의하여 정할 것이 아니고 특허청구범위와 특허청구범위의 균등한 것에 의해 정해 져야 한다.

### 【발명의 효과】

<37> 상술한 바와 같이 본 발명은 키 매트릭스상의 키 상태 변화로부터 키 인터럽트가 발생하면 키 인터럽트 결과 값 그 자체만으로도 어떤 키의 변화가 키 인터럽트를 발생시켰는지 산술적으로 계산하여 알아낼 수 있다. 그 결과 키 입력에 따른 키 위치 값을 쉽게 인식할 수 있다.

**【특허청구범위】****【청구항 1】**

키 매트릭스 구조를 가지는 단말장치에서의 키 인식방법에 있어서,

상기 키 매트릭스 상의 이전 키 상태 값과 키 상태 변화에 따른 현재 키 상태 값과의 차이를 이용하여 키 누름과 키 떨어짐을 파악하는 과정과,

상기 키 매트릭스를 구성하는 행들과 열들 각각에 대응 비트가 할당되어 있고, 상기 행들 각각에서 인접한 열들간 차이 값과 상기 열들 각각에서의 인접한 행들간 차이 값을 고려한 관계식과 상기 이전 키 상태 값과 현재 키 상태 값간의 차이를 이용하여 키 매트릭스 상에서의 입력 키 위치 값을 구하는 과정으로 이루어짐을 특징으로 하는 키 인식 방법.

**【청구항 2】**

제1항에 있어서, 상기 관계식은  $2^{j+5} + 2^i$ 임( $i, j$ 는 변수로서 키 매트릭스의 행과 열을 가리킴)을 특징으로 하는 키 인식방법.

**【청구항 3】**

제1항에 있어서, 상기 키 누름으로의 파악은 상기 현재 키 상태 값에서 이전 상태 값을 뺀값이 양수일 때이고, 상기 키 떨어짐으로의 파악은 상기 현재 키 상태 값에서 이전 키 상태 값을 뺀 값이 음수 일 때임을 특징으로 하는 키 인식방법.

## 【청구항 4】

키 매트릭스 구조를 가지는 단말장치에서의 키 인식방법에 있어서,

상기 키 매트릭스 상의 이전 키 상태 값과 키 상태 변화에 따른 현재 키 상태 값과의 차이를 이용하여 키 누름과 키 떨어짐을 파악하는 과정과,

상기 현재 키 상태 값 - 이전 키 상태값 =  $2^{j+5} + 2^i$  관계를 이용해서 변수  $i, j$ 를 계산하는 과정과,

상기 변수  $i, j$ 로 키 매트릭스 상에서 입력 키 위치 값을 구하는 과정으로 이루어짐을 특징으로 하는 키 인식방법.

## 【청구항 5】

제4항에 있어서, 상기 입력 키 위치값은  $key[i][j]$ 임(여기서  $i$ 는 키 매트릭스의 행의 위치이고  $j$ 는 키 매트릭스의 열의 위치)을 특징으로 하는 키 인식방법.

## 【청구항 6】

키 매트릭스 구조를 가지는 단말장치에서의 키 인식방법에 있어서,

상기 키 매트릭스 상의 이전 키 상태 값과 키 상태 변화에 따른 현재 키 상태 값이, 상기 현재 키 상태 값 - 이전 키 상태값 =  $2^{j+5} + 2^i$ 가 되는 관계를 이용해서 변수  $i, j$ 를 계산하는 과정과,

상기 변수  $i, j$ 로 키 매트릭스 상에서 입력 키 위치 값  $key[i][j]$ 임(여기서  $i$ 는 키 매트릭스의 행의 위치이고  $j$ 는 키 매트릭스의 열의 위치)을 구하는 과정으로 이루어짐을



특징으로 하는 키 인식방법.

【청구항 7】

키 매트릭스 구조를 가지는 단말장치에서의 키 인식방법에 있어서,

상기 키 매트릭스 상에서 키 상태 변화에 따른 제1 키 상태 값을 제1 임시 저장부에 저장하는 과정과,

상기 제1 키 상태 값 저장 이후 키 상태 변화에 따른 제2 키 상태 값을 제2 임시 저장부에 저장하는 과정과,

상기 저장된 제1 키 상태 값과 제2 키 상태 값과의 차이를 이용하여 상기 키 매트릭스상에서의 키 누름과 떨어짐을 파악하는 과정과,

상기 현재 키 상태 값 - 이전 키 상태값 =  $2^{j+5} + 2^i$  관계를 이용해서 변수  $i, j$ 를 계산하는 과정과,

상기 변수  $i, j$ 를 이용해 키 매트릭스 상에서의 입력 키 위치 값을 구하는 과정으로 이루어짐을 특징으로 하는 키 인식방법.

【청구항 8】

제7항에 있어서, 상기 입력 키 위치값은  $key[i][j]$ 임(여기서  $i$ 는 키 매트릭스의 행의 위치이고  $j$ 는 키 매트릭스의 열의 위치)을 특징으로 하는 키 인식방법.

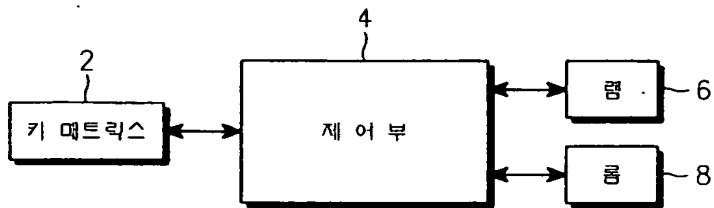
【청구항 9】

제7항에 있어서, 상기 입력 키 위치 값을 구한 후 상기 제2 키 상태값을 제1 임시  
【청구항 9】  
저장부에 저장하는 과정을 더 가짐을 특징으로 하는 키 인식방법.



【도면】

【도 1】

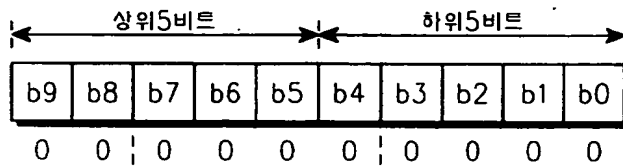


【도 2】

	b5	b6	b7	b8	b9
b0	key[0][0]	key[0][1]	key[0][2]	key[0][3]	key[0][4]
b1	key[1][0]	key[1][1]	key[1][2]	key[1][3]	key[1][4]
b2	key[2][0]	key[2][1]	key[2][2]	key[2][3]	key[2][4]
b3	key[3][0]	key[3][1]	key[3][2]	key[3][3]	key[3][4]
b4	key[4][0]	key[4][1]	key[4][2]	key[4][3]	key[4][4]



【도 3】



【도 4】

